Contents
Introduction........................................................................................................................................1
What you will learn in this tutorial....................................................................................................2
Lesson 1: Create First Form Using Wizard......................................................................................3
Lesson 2: Design the Second Form...................................................................................................9
   Add Components to Panel.............................................................................................................10
   Back to the ‘List Persons’ .............................................................................................................16
   Edit Person....................................................................................................................................18
   List Persons....................................................................................................................................18
   Input Variables .............................................................................................................................20
HTML Templates................................................................................................................................21
Execute the application as Swing native ...........................................................................................22
DbSchema Management Applications.................................................................................................23
End....................................................................................................................................................23

Introduction

The Forms and Reports designer is tool inside DbSchema. Use it to design master-details reports as well as small applications for Web or Swing. They are innovative:

- The generated reports are compatible with mobile devices and tables by using responsive bootstrap HTML templates.
- The reports are organized hierarchical, allowing an unlimited number of master-detail sections.
- The reports can use of custom logic based on Java-compatible Groovy language.

The designer is different from what you may know from other tools. Read this tutorial to understand how they work.
What you will learn in this tutorial

We will build together a form and a report. We will create components and add custom logic to them. We will use the same designer for forms as well as reports, as they are similar. The report has no interactivity and uses a different HTML template.

We will build an interactive page showing the content of the ‘person’ table, with navigation and edit buttons.

<table>
<thead>
<tr>
<th>Title</th>
<th>Firstname</th>
<th>Lastname</th>
<th>Email</th>
<th>Birthdate</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dr.</td>
<td>Abel</td>
<td>Spencer</td>
<td><a href="mailto:cupo@i-ow.org">cupo@i-ow.org</a></td>
<td>2014-08-12</td>
<td>269 North White Nobel Parkway</td>
</tr>
<tr>
<td>Ms.</td>
<td>Janice</td>
<td>Montes</td>
<td><a href="mailto:jjg1@--jn.com">jjg1@--jn.com</a></td>
<td>2018-05-17</td>
<td>644 North Green Old Parkway</td>
</tr>
<tr>
<td>Dr.</td>
<td>Gretchen</td>
<td>Proctor</td>
<td><a href="mailto:vxxd.xg@e--eh.org">vxxd.xg@e--eh.org</a></td>
<td>2016-03-19</td>
<td>774 East Green Second Freeway</td>
</tr>
<tr>
<td>Dr.</td>
<td>Lawanda</td>
<td>Valazquez</td>
<td><a href="mailto:cppld@li--jl.net">cppld@li--jl.net</a></td>
<td>2018-05-16</td>
<td>60 West Green Second Freeway</td>
</tr>
<tr>
<td>Dr.</td>
<td>Robbie</td>
<td>Wilkins</td>
<td><a href="mailto:vloj@---vg.com">vloj@---vg.com</a></td>
<td>2004-05-09</td>
<td>92 South Green New Avenue</td>
</tr>
<tr>
<td>Dr.</td>
<td>Carla</td>
<td>Randall</td>
<td>odth233@p---com</td>
<td>2013-08-04</td>
<td>647 North Rocky Old Avenue</td>
</tr>
<tr>
<td>Mr.</td>
<td>Kenda</td>
<td></td>
<td></td>
<td>2001-07-26</td>
<td>548 North Rocky New Boulevard</td>
</tr>
</tbody>
</table>

Pressing the edit button will open another page for editing the person name.

Data can be edited

Page is navigable

Edit data and submit the changes into the database.
Lesson 1: Create First Form Using Wizard

For creating our forms we will use the sample project from DbSchema. Start DbSchema and close any open project (if any). The welcome screen will show up. Choose the sample project.

The sample project is connected to an embedded H2 database. The interface should look like below.
Right-click the table ‘PERSONS’ and choose ‘New Form or Report’ from pop-up.

A wizard dialog should appear. Type ‘List Persons’ in the form name field, keep the other settings as they are and press the ‘Continue’ button.
In the next screen will open the forms designer on top and a query builder on the bottom. We will use the query builder to create a query for the displayed data.

In the query builder select all checkboxes (right-click the table header for an option). Run the query and the result pane will show up. From the result pane choose ‘Continue Wizard’.

1) Run Query
2) Press ‘Continue Wizard’
The dialog below should appear. Here we can choose a component to place for each of the database fields. Leave the ‘personid’ and ‘clubid’ unselected and choose labels for all the others.

Our first form is ready. The form will look like below.

Let’s execute the form by pressing the play button from the menu.
Let’s compare the form in the designer and the output we got. The form is build out of panels (the rounded corner rectangles). We have three panels: navbar, page and table panels. Each panel contains one or more cells.

The navbar is for menu components, now is empty and won’t show in the output.

The page panel is the page itself, the header and first, previous and next buttons on the bottom belongs to it. The table panel is showing the data.
The table panel has associated a data source script which is the SQL query created using the query builder. The data source is returning the rows which are getting displayed as a table in the output. The first row in the panel with the blue background is a header row and will show only one time in the output. The second row is a body row and will repeat for each data record in the output.
Lesson 2: Design the Second Form

In this lesson we will build the second form called ‘Edit Person’. This form will be used for editing any of the persons first and last name. The form will show as dialog on top of the previous form. We will create the components by ourselves, will add custom logic to the form and interconnect the two forms using buttons. The form will look like bellow.

To create it choose from the DbSchema main menu ‘Forms’ the option ‘New Form or Report’.

The wizard will show up. Enter form name ‘Edit Person’, select ‘Skip Editing’ option and press continue.
An empty form is created. The page panel has 12 columns because is preset to use a bootstrap grid template. The bootstrap grid used for responsive pages compatible with mobile devices is using twelve (12) columns of equal size.

Add Components to Panel

We will add components to panel one by one, without wizard. For this click the first cell and press the label button in the menu. Alternative you can right-click the cell you can choose the label from the pop-up menu.

The label dialog will show up. Enter ‘First Name’ in the text field and press ‘Ok’.
The label is created:

The label editor can be reopened by double-clicking the label.

The label can be moved to another cell by drag and drop of the label text.

One component can cover multiple cells. Drag and drop the grip on the bottom-right side to resize it.

Just in few steps you learned how to create a component, move and resize it.

Now the label should cover 3 cells. Let’s continue and create a text field in the next cell. Click the cell and choose the text field from the menu.
Type ‘firstname’ in the ‘id’ field of the text field editor.

Drag & drop from the bottom grip to resize the text field to cover all left cells.

Next we will add label and text field for the last name as well. For this we are going to add one more row to the panel. Right-click the text field and choose ‘Insert Row or Column’.
Choose ‘Insert row down’ to insert the row below the current cell row.

The row is created.

Now create one more label and text field for the last name. Use ‘lastname’ for the text field id. Resize them accordingly.
Add one more row and create a button.

Use ‘Save’ as button text.

In the action field we can choose what will be done when the button is clicked. Choose to open the ‘List Persons’ form when the button is clicked.

We want to inform as well the ‘List Persons’ form that an editing was done, so that the ‘List Persons’ will apply the editing in the database. In HTML the database is first being updated first in the next page called after the edit page. This is the ‘List Persons’ page, and a script inside this page will check for the ‘operation’ variable to know if an edit was done or not. Therefore type in the variables the text ‘operation=edit’. Later we will explain variables in detail.

Press ‘Ok’. Now we are done with the components.

We want to open the ‘Edit Persons’ as a dialog on top of the ‘List Persons’. To do this we have to edit the form properties.
The form properties dialog will open. Here choose ‘Dialog’ for the ‘view in’ field and for the template as well. Press ‘Ok’.

At this point we should have created two forms. They should be both visible as distinct tab in the current layout.
If you close any of the tabs they can be later reopen the forms from the application menu. All created forms will be saved in DbSchema project file.

Similar you can create different other components from the menu.

**Back to the 'List Persons'**

At this point we will create a button in the ‘List Persons’ form to access the ‘Edit Person’.

Open the ‘Edit Person’ form, add one column to the right and place a hyperlink button inside.
Type ‘Edit’ in the button text field. On click choose to open the ‘Edit Person’ form and in the variables field enter the text below:
lastname=${lastname}&firstname=${firstname}&personid=${personid}

The application is using variables to send data between pages. A variable has a name and a value. DbSchema is using a similar way as in HTML URLs to communicate data between pages.

The application will replace the ${…} with the given variable names and send it to the target form.

💡 Variables provided by input fields are automatically sent to the next form

We added here the firstname, lastname and personid because they are provided by the data source and there is no input field to store them in the interface.
If our form would contain a text field, combo, radio, list, etc. with one of these variables, there would be no need to add them here. The values provided by one of this input fields are send automatically to the next form.
Remember that labels do not belong to input fields; they only show some data but are not sent between pages.

💡 Variable names are case sensitive
Look the image below to see which variables are sent between pages.

We will implement the logic to apply the changes in the database in the initialization script of the 'List Persons'. Press the form title to access the form properties dialog.

Here click the ‘Initialization script’ button.
The Groovy editor will open. Copy & paste the text below into the editor.

```groovy
if ( 'edit'.equals( operation )){
    println "Going to update ${personid}"
    sql.executeUpdate( "UPDATE persons SET firstname=?, lastname=? WHERE personid=?", [firstname, lastname, personid] )
    sql.commit()
}
```

The script is checking if the variable ‘operation’ has the value ‘edit’, and if this is true a database update is executed. The ‘sql’ is the connection to the database.

You can close the Groovy editor by pressing the ‘x’ on the tab, the text is auto-saved into the project.

Now is the time to test the created pages. Press ‘run’ in the ‘List Persons’ form.
In the initialization script there is a `println` command. We use this to debug the script. The command output will show in the DbSchema Output Logs dialog. The dialog pop-ups automatically when the `println` is reached.

💡 All script exceptions are shown in the ‘Output Logs’ dialog accessible from the ‘Help’ main menu.

**Input Variables**

Input variables are declared in the form properties dialog. Two input variables are already created in the ‘List Persons’ form properties dialog.
The input variables `listPersonsPageNumber` and `listPersonsRecordsPerPage` were created by the wizard in first lesson. They are used to implement the pagination.

Double-click any of them to edit.

Input variables are used to ensure a certain variable is present when the form is open or to set-up a variable default value if is not present.

![Edit Variable](image)

Each panel with a data source script is checking if the variables `<panelid>PageNumber` and `<panelid>RecordsPerPage` exists. If yes then the data source will consider showing only the given number of records and will scroll to the given page.

**HTML Templates**

The form and each component apart have its own HTML template. The templates are generating the HTML. This means you can modify the look of a component or you can add new representation for a component. Each component can have multiple templates. All templates together are groped in a bundle and your application can choose the bundle to use.

The template manager is accessible from the DbSchema main menu.

![Template Manager](image)

A detailed presentation of the template manager is done in the application help.
**Execute the application as Swing native**

DbSchema can execute the created forms as swing applications as well. Switch to swing from the designer menu.

For Swing the designer will show small arrow icons on top of each column or row. They are responsible for setting the space distribution between columns when the window is resized. The left arrow means the column is taking minimal space to fit the components inside. The right arrow means the column is ‘filling’ the space, which means it takes all the space left in the window when the window size increase. Components like text areas placed inside will get more space when the window size is increased. A special Swing table layout is used to implement this.

Please notice some components as the table in the image above are rendered as HTML even if the main dialog is Swing. The table panel has a setting for this, allowing to embed HTML in Swing.
DbSchema Management Applications

Based on the forms and reports module we started implementing database management applications showing information about the database space usage, database activity and locks, management of user roles and rights, etc. The applications are now available from the forms menu (now available only for MySql).

The management applications are open source. Please feel free to contribute on development. Please write us on support@dbschema.com about this.

End

The forms and reports are a new in DbSchema 7. Please help us to improve them by writing us back on www.dbschema.com support page.